REMARKS

Claims 1-23 are pending. Claims 1-23 stand rejected. Reconsideration and allowance of the above-referenced application are respectfully requested.

The specification has been amended to correct discovered informalities. No new matter is added by this Amendment. Enclosed is a marked-up version of the changes being made to the specification by the current Amendment, and substitute pages 2 and 9 of the specification with these changes incorporated.

The title stands objected to as not descriptive. The title has been amended to further clarify the subject matter of the present application.

Claims 1-4, 6-7, 10-14, and 20-23 stand rejected under 35 U.S.C. 102(a) as allegedly being anticipated by U.S. Patent No. 5,832,219 (Pettus). This rejection is respectfully traversed. Claims 5, 8-9, and 15-19 stand rejected under 35 U.S.C. 103(a) as allegedly being unpatentable over Pettus in view of U.S. Patent No. 6,263,491 (Hunt). This rejection is respectfully traversed.

The claims define systems and techniques whereby a client can generate activation requests to be fulfilled by a server, even if the client lacks information about any specific server that can process such requests. This allows client nodes to

create remote components on available server nodes without monitoring the state of the network. This is supported by the specification, for example, on page 2, line 13, to page 3, line 3. The art of record fails to teach or suggest these systems and techniques as claimed to promote the above-described advantage.

With respect to independent claim 1, the server node enables "the client node to activate remote components on available server nodes without specific names or capabilities of nodes in the network servicing the requests." With respect to independent claim 6, the first and second modules enable "the client to trigger creation of remote components without specific names or capabilities of network nodes servicing that creation." With respect to independent claim 23, the client nodes are able "to request activation of remote components at run-time without specific names or capabilities of nodes servicing those requests."

None of the art of record, including Pettus, either teaches or suggests this aspect of the claims. In fact, the client node of Pettus necessarily knows the specific name of the server node servicing the request. (See figure 6 and the accompanying description.) The invention described in Pettus uses a dispatcher on a known server to handle client requests. (See

col. 5, lines 28-40.) Thus, it is respectfully suggested that independent claims 1, 6 and 23 should be allowable.

With respect to independent claims 7, 14, 20 and 22, the art of record, including Pettus, fails to teach or suggest multicasting a machine-independent activation request to the network as claimed. With respect to independent claims 10 and 21, the art of record, including Pettus, fails to teach or suggest monitoring at a server a specific port to receive a machine-independent client activation request and returning capability information of the server to the client. Thus, it is respectfully suggested that independent claims 7, 10, 14, 20, 21 and 22 should be allowable.

With respect to claims 2-5, 8-9, 11-13 and 15-19, each of these claims depends from an allowable base claim for the reasons discussed above. As such, it is respectfully suggested that these claims should be allowable.

In view of the above amendments and remarks, therefore, all of the claims should be in condition for allowance. A formal notice to that effect is respectfully solicited. Attached is a marked-up version of the changes being made by the current Amendment.

No fee is believed due with this Response. Please apply any other charges or credits to Deposit Account No. 06-1050.

Respectfully submitted,

Date: Feb. 27, 2002

William E. Hunter
Reg. No. 47,671

Fish & Richardson P.C.
PTO Customer No. 20985
4350 La Jolla Village Drive, Suite 500
San Diego, California 92122
Telephone: (858) 678-5070
Facsimile: (858) 678-5099

10166422.doc

Version with markings to show changes made

In the specification:

The title has been amended as follows:

-- DISTRIBUTED COMPONENT SYSTEM MANAGEMENT USING MACHINE-INDEPENDENT ACTIVATION REQUESTS --

Paragraph beginning at page 1, line 19 has been amended as follows:

-- Since DCOM activation requests depend on the client being aware of the name and/or IP address of a specific server, the client often has either to monitor the detailed state of the network at the time of the request or to assume the server is available and configured properly to service the request. Server failures become evident to the client only after the activation request has been committed to the server by the client [**to the server by the client [CORRECT?]**], at which time it may be too late for the client to mitigate the problem. There is often no mechanism available for the client to dynamically attempt connections with other anonymous and viable nodes in response to a failure of the currently used server because of the static nature of a DCOM based distributed system. At best, a response to the server failure often requires informing application users based on the network configuration,

and at worst, it may require a complete recompilation of source code. --

Paragraph beginning at page 8, line 18 has been amended as follows:

-- In the Multi-Ci mode, the parameters for the client request include a maximum response wait time as as well as maximum and minimum response count just as with the SNR mode, but the returned values will instead be the interface pointers requested. The IP augmentation module for the client node 202 creates location independent references to objects on the network by using an existing DCOM protocol known as an Object RPC (ORPC). **[[,--WE SHOULD DESCRIBE THIS PROCESS IN DETAIL. I TRIED TO WRITE OUT THE PROCESS (TWO PARAGRAPHS BELOW; IT MAY NOT MAKE SENSE). PLEASE ADD, CORRECT OR DELETE AS YOU SEE FIT.]]**The ORPC is a set of definitions that extends the standard DCE RPC protocol. It specifies how calls are made across the network and how references to objects are represented and maintained. --

The ORPC is a set of definitions that extends the standard DCE RPC protocol. It specifies how calls are made across the network and how references to objects are represented and maintained.

ORPC uses standard RPC packets, with additional DCOM specific information, in the form of an interface pointer identifier, conveyed as additional parameters on calls and replies. The interface pointer identifier is used to identify a specific interface on a specific object on a server machine where the call will be processed.

One of the parameters of an activation response packet is the marshaled interface pointer which is represented in an object reference (OBJREF) structure. The OBJREF structure is a data type that represents a reference to an object and contains a signature field of hex value 0x5747454D. This sequence, which reads 'MEOW' in ASCII, is useful when scanning for the object reference packet.

A flow diagram of the IP augmentation module for the server system 202 is shown in FIG. 6. The system 202 monitors and listens on a specific port that is tied to the multicast IP address, at step 600. Again, the server may service the